

L^AT_EX Example Document

Chris Eppolito

2nd September 2022

This is a sample document to show you how to use L^AT_EX to write nice documents for your homework submission. I have built the document to get you started.

Notice that the text is double-spaced. This is on purpose! I expect the paper to have enough space for me to mark it with comments and return it to you.

1. How much wood could a woodchuck cut if a woodchuck could cut wood?

Once you have stated a problem, you can solve it below using the `solution` environment.

Solution: This is my solution to the first question. When I write maths, I always enclose it between `\(CONTENTS \)`. This way L^AT_EX knows I mean to write in-line maths. Something like $e^{i\pi} + 1 = 0$ is in-line maths.

To write display-style maths, I enclose it between `\[CONTENTS \]`. This way L^AT_EX knows I mean to write display-style maths. Something like

$$e^{i\pi} + 1 = 0$$

is display-style maths.

$$\sum_{k=1}^n k^2 = \frac{n(n+1)(2n+1)}{6} = \frac{1}{6}n(n+1)(2n+1)$$

To comment in the file, use a `%` symbol. Everything else on that line will be ignored...

For multi-line display-style maths, you will need to make friends with `align*`. This environment uses the symbol `&` for alignment, and uses `\\` for line-breaks. Have a look at the code for the following to see what I mean.

$$\begin{aligned} S &= \left\{ n \in \mathbb{N} : n + \sum_{k=1}^5 1 \leq 50 \right\} = \{n \in \mathbb{N} : n \leq 45\} \\ &= \{n \in \mathbb{Z} : 0 \leq n \leq 45\} \end{aligned}$$

Notice that the symbols `\` and `{` and `}` are special in \LaTeX . Every command begins with a `\`, and the `.tex` file won't compile if you have unbalanced braces because they are used to determine limited scopes. To get the backslash for set difference, type `\setminus`, and to get the braces for sets, type `\{` and `\}`. This makes an expression like `[n] \ S = \{1, 4, 7, n\}` correctly.

To end your proofs with a tombstone (like I do in class), write `\qed` at the end. \square

Some commands only make sense in math-mode (i.e. in one of the `\(\)`, `\[\]`, or `\begin{align*}\end{align*}` environments). To make subscripts use an underscore `_`, and to make superscripts use a caret `^`. These are treated differently depending on the math-mode! The command `\sum_{k=0}^n \binom{n}{k} = 2^n` looks like $\sum_{k=0}^n \binom{n}{k} = 2^n$ in-line, but like so in display-style.

$$\sum_{k=0}^n \binom{n}{k} = 2^n$$

You might also like to use a table (e.g. for a truth table). You can use an `array`

environment to make this happen. Here is the truth table for implication.

P	Q	$P \rightarrow Q$
T	T	T
T	F	F
F	T	T
F	F	T

Note we used the `\hline` command to build the horizontal rule line and the vertical bar `|` in the argument of the `array` to separate the `c` characters; this tells \LaTeX we plan to have centered (i.e. `c`) columns and vertical rulings between them. You need as many `cs` as you have columns in those braces.

Using [Overleaf](#) makes writing \LaTeX fairly straightforward. It comes with many helpful features. If you run into problems, search on Overleaf for the answer.

Using [Detexify](#) is good for finding symbols (e.g. γ or an α). There you just handwrite the symbol you want, and the site pattern-matches your symbol to find the command.